COLLABORATING WITH HUMANOID ROBOTS IN SPACE

DONALD SOFGE[1], MAGDALENA BUGAJSKA[1], J. GREGORY TRAFTON[2],
DENNIS PERZANOWSKI[1], SCOTT THOMAS[1], MARJORIE SKUBIC[3],
SAMUEL BLISARD[4], NICHOLAS CASSIMATIS[2], DEREK BROCK[2],
WILLIAM ADAMS[1], ALAN SCHULTZ[1]

[1,2]Navy Center for Applied Research in Artificial Intelligence,
Naval Research Laboratory
Washington, DC 20375, USA
[1]{sofge, magda, dennisp, thomas, adams, schultz}@aic.nrl.navy.mil
[2]{trafton, cassimatis, brock}@itd.nrl.navy.mil

[3,4]Electrical and Computer Engineering Department
University of Missouri-Columbia
Columbia, MO 65211, USA
[3]skubicm@missouri.edu, [4]snbfg8@mizzou.edu

One of the great challenges of putting humanoid robots into space is developing cognitive capabilities for the robots with an interface that allows human astronauts to collaborate with the robots as naturally and efficiently as they would with other astronauts. In this joint effort with NASA and the entire Robonaut team we are integrating natural language and gesture understanding, spatial reasoning incorporating such features as human-robot perspective taking, and cognitive model-based understanding to achieve a high level of human-robot interaction. Building greater autonomy into the robot frees the human operator(s) from focusing strictly on the demands of operating the robot, and instead allows the possibility of actively collaborating with the robot to focus on the task at hand. By using shared representations between the human and robot, and enabling the robot to assume the perspectives of the human, the humanoid robot may become a more effective collaborator with a human astronaut for achieving mission objectives in space.

Keywords: Autonomous Systems; Humanoid Robot; Cognitive Model; Spatial Reasoning.

1. Introduction

As we develop and deploy advanced humanoid robots such as Robonaut,[1] NASA's robotic astronaut assistant platform, to perform tasks in space in collaboration with human astronauts, we must consider carefully the needs and expectations of the human astronauts in interfacing and working with these humanoid robots. We want to endow the robots with the necessary capabilities for assisting the human astronauts in as efficient a manner as possible. Building greater autonomy into the robot will diminish the human burden for controlling the robot, and making the humanoid robot a much more useful collaborator for achieving mission objectives in space.

In this effort we build upon our experience in designing multimodal human-centric interfaces and cognitive models for dynamically autonomous mobile robots. We argue that by building human-like capabilities into Robonaut's cognitive processes, we can achieve a high level of interactivity and collaboration between human astronauts and Robonaut. Some of the necessary components for this cognitive functionality addressed in this paper include use of cognitive architectures, natural language and gesture understanding, and spatial reasoning with human-robot perspective-taking.

Portions of this work published previously as:
D. Sofge, D. Perzanowski, M. Skubic, N. Cassimatis, J. G. Trafton, D. Brock, M. Bugajska, W. Adams, and A. Schultz, Achieving Collaborative Interaction with a Humanoid Robot, in *Proceedings of the Second International Conference on Computational Intelligence, Robotics and Autonomous Systems (CIRAS)* (December 2003).

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
2005		**00-00-2005 to 00-00-2005**

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
Collaborating with Humanoid Robots in Space	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Naval Research Laboratory,Navy Center for Applied Research in Artificial Intelligence (NCARAI),4555 Overlook Avenue SW,Washington,DC,20375**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES
International Journal of Humanoid Robotics. 2(2), 181-201

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**Same as Report (SAR)**	**20**	

2. Cognitive Architectures for Robots

Most of Robonaut's activities involve interaction with human beings. We base our work on the premise that embodied cognition, using cognitive models of human performance to augment a robot's reasoning capabilities, facilitates human-robot interaction in two ways. First, the more a robot behaves like a human being, the easier it will be for humans to predict and understand its behavior and interact with it. Second, if humans and robots share at least some of their representational structure, communication between the two will be much easier. For example, both in language use[2] and other cognition[3], humans use qualitative spatial relationships such as "up" and "north". While it would not be impossible, it would be difficult, and probably highly unnatural, to interact with a robot using only real number matrices. Humans employ spatial relationships and utilize qualitative transformations to express them. Therefore, to facilitate communication, we believe it is necessary to endow the robot with qualitative representations of space parallel to those utilized by humans. In previous efforts we have used cognitive models of human performance of tasks to augment the capabilities of robotic systems.[4,5]

We have investigated the use of two cognitive architectures based on human cognition for certain high-level control mechanisms for Robonaut. These cognitive architectures are ACT-R[6] and Polyscheme.[7] ACT-R is one of the most prominent cognitive architectures to have emerged in the past two decades as a result of the information processing revolution in the cognitive sciences. Recognized as a unified theory of cognition, ACT-R is a relatively complete theory about the structure of human cognition that strives to account for the full range of cognitive behavior with a single, coherent set of mechanisms. Its chief computational claims are: first, that cognition functions at two levels, one symbolic and the other subsymbolic; second, that symbolic memory has two components, one procedural and the other declarative; and third, that the subsymbolic performance of memory is an evolutionarily optimized response to the statistical structure of the environment. These theoretical claims are implemented as a production-system modeling environment. The theory has been successfully used to account for human performance data in a wide variety of domains including memory for goals,[8] human computer interaction,[9] and scientific discovery.[10] We will use ACT-R to create cognitively plausible models of appropriate tasks for Robonaut to perform.

We use Cassimatis' Polyscheme[7] architecture for spatial, temporal and physical reasoning. The Polyscheme cognitive architecture enables multiple representations and algorithms (including ACT-R models), encapsulated in "specialists", to be integrated into inference about a situation. We use an updated version of the Polyscheme implementation of a physical reasoner to help keep track of Robonaut's physical environment.

2.1. *Perspective-taking*

One feature of human cognition that is very important for facilitating human-robot interaction is "perspective-taking". There is extensive evidence that human perspective-taking develops in young children around the age of four to five.[11,12,13] In order to understand utterances such as "the wrench on my left", the robot must be able to reason from the perspective of the speaker what "my left" means.

To explore how and when people use perspective taking (especially spatial perspective taking) in a context relevant to a robot like Robonaut, we examined two astronauts working at the neutral buoyancy laboratory (NBL) at NASA/JSC. In the NBL, astronauts conduct a wide variety of training for extravehicular activity (EVA); i.e., working outside the space shuttle, including working out the procedures and defining roles to perform EVAs.

As part of this project, the following conversation (Table 1) occurred between three individuals — two astronauts (EV1 and EV2) in the neutral buoyancy tank — and one person (Ground) outside of the tank in mission control. The latter watched the two astronauts through a video feed of the activity.

Table 1: Dialog between two astronauts and an observer.

EV1	EV2	Ground
		Bob, if you come straight down from where you are, uh, and uh kind of peek down under the rail on the nadir side, by your right hand, almost straight nadir, you should see the uh,
	Mystery hand-rail	
		The mystery hand-rail, exactly
	OK	
There's a mystery hand-rail?		
		Oh, it's that sneaky one. It's there's only one in that whole face.
Oh, yeah, a mystery one.		
		And you kinda gotta cruise around until you find it sometimes.
I like that name.		

Notice several things about this conversation. First, the mission control person mixes reference frames from addressee-centered ("by your right hand") and exocentric ("straight nadir" which means towards the earth) in one instruction, the very first utterance. Second, the participants come up with a new name for a unique unseen object ("the mystery hand-rail") and then tacitly agree to refer to it with this nomenclature later in the dialog.

This short excerpt shows that an automated reasoning system needs to be able not only to mix perspectives, but to do so in a rather sophisticated manner. One of the most difficult aspects of this problem is the addressee-centered point of view, which happens quite often in the corpus we have examined. Thus, in order for a robotic system to be truly helpful, it must be able to take into account multiple perspectives, especially another person's perspective. There are, of course, multiple ways of solving this problem. We have worked on two possibilities, both focusing on computational cognitive models. Both models have been described in more detail elsewhere.[14,15,16]

2.1.1. *Perspective taking using similar processes: Polyscheme*

Polyscheme is a cognitive architecture based on the ability to conduct mental simulations of past, future, distant, occluded and/or hypothetical situations. Our approach has been to use Polyscheme to enable robots to simulate the world from the

perspective of people with whom they are interacting, and to understand and predict the actions of humans.

Polyscheme uses several modules, called specialists, which use specialized representations for representing some aspect of the world. For example, Polyscheme's space specialist uses cognitive maps to represent the location of and spatial relations among objects. Its physics specialist uses causal rules to represent the causal relationship between events. Using these specialists, Polyscheme's specialists can simulate, i.e., represent the state and predict subsequent states, of situations it cannot see at present, either because they occurred in the past or future, they are occluded from view and/or they are hypothetical.

Polyscheme modelers have the ability to set strategies for choosing which situations to simulate in what order. Modelers use these strategies to implement reasoning and planning algorithms, including perspective taking. For example, the counterfactual simulation strategy "when uncertain about A, simulate the world where A is true and the world where A is false" implements backtracking search when used repeatedly. The stochastic simulation strategy "when A is more likely to be true than false, simulate the world where A is true more often than the world where A is false" implements an approximate form of probabilistic reasoning (often used, e.g., to estimate probabilities in a Bayesian network). Polyscheme's ability to combine multiple simulations from multiple strategies and to share simulations among strategies is the key to tightly integrating multiple reasoning and planning algorithms.[14] Since each simulation is conducted by specialists that use multiple representations (e.g., perceptual, spatial, etc.), the integration of reasoning with sensation and multiple forms of reasoning is inherent and on-going.

Fig. 1. The robot needs to take the perspective of the person in order to determine to which cone the human has referred.

By using Polyscheme to implement the perspective simulation strategy "when a person, P, takes action, A, at time, T, simulate the world at time T from A's perspective," we have given our robots the ability to reason about the world from the perspective of people and to thereby disambiguate their utterances. In many cases, for instance, an utterance is ambiguous given the listener's knowledge, but unambiguous given the speaker's knowledge. Figure 1 is an example. The figure shows a robot and a person facing each other. The robot can see that there are two cones in the room, cone1 and cone2, but the person only knows about cone2 because cone1 is hidden from her. When the person commands, "Robot, go to the cone", the phrase "the cone" is potentially ambiguous to the robot because there are two cones, though unambiguous to the person because she only knows of the existence of one cone. Intuitively, if the robot could take the perspective of the person in this task, it would see that, from that perspective, cone2 is the only cone and therefore "the cone" must refer to cone2.

We have used Polyscheme to implement this sort of reasoning on Robonaut. Generally, Polyscheme uses perspective taking and mental simulation to determine which cone the person can see (creating multiple hypothetical worlds until the "correct" world is found and the correct cone then becomes disambiguated for the robot.[14]

2.1.2. *Perspective taking using similar representations: ACT-R*

The cognitive architecture jACT-R is a java version of the ACT-R architecture.[17] To represent declarative memory, it uses chunks of various types of elements. These chunks can be accessed through a memory retrieval buffer. In order to use and manipulate the chunks of memory, ACT-R provides a framework for production rules. A sample chunk and production rule is shown in Figure 2. ACT-R then simulates cognitive behavior and thought based on activation values and propagation of chunks and higher-level goals. ACT-R also includes support for perceptual and motor cognitive tasks by including a second visual buffer for viewing objects in space.

ACT-R/S extends jACT-R to implement a theory about spatial reasoning.[18] It posits that spatial representations of objects are temporary, egocentric and dynamically updated.[19] ACT-R/S has three buffers for spatial cognition: the configural buffer, the manipulative buffer, and the visual buffer. The configural buffer represents spatial extents of objects that are updated during self-locomotion and is used during navigation, path-computation, object-avoidance, etc. The manipulative buffer represents the metric spatial bounds of an object and is used for spatial transformations of objects.[20,16] The visual buffer is the same as the "standard" perceptual-motor buffer in ACT-R/PM.[21]

ACT-R/S represents objects using vectors to the visible sides of the object. It has the ability to track these objects through the configural buffer, a data structure analogous to the other buffers of ACT-R that stores each object once it has been identified. The coordinate vectors of the objects in the buffer are then dynamically updated as the agent moves throughout the spatial domain. The configural buffer, unlike the visual and retrieval buffers of ACT-R, can hold more than one object to account for the fact that animals have been shown to track more than one landmark at once while moving through the world.[22] In order to focus on the representational aspects of perspective-taking, we have built our model using only the spatial representations within jACT-R/S.

Using the configural extension begins with locating and attending to an object via the visual buffer provided by the standard Perceptual-Motor extension to ACT-R. Once an object is found, it is possible to request that the ACT-R/S-visual object at that location, if one exists, be placed in the configural buffer. The model then begins tracking this object, creating the initial vectors and updating them as the agent moves around in the world. The updating transformation is done by adding or subtracting vectors representing the agent's movement to the vectors and object's location.

```
chunk_cone:
 isa: cone
 color: gray
 speaker_can_see: true
 location: (x,y)

production_take_cone:
 if isa cone
        and speaker_can_see
        and (my_x, my_y) = (x,y)
 then take_cone
```

Fig. 2. An ACT-R memory chunk and production rule.

In order to demonstrate the results of perspective taking using jACT-R/S, we solved the same perspective-taking task that Polyscheme did: disambiguating which cone a person referred to when the robot could see two cones but the person could only see one. For this example, we did not implement the full system on a physical robot. In the simulated world, two agents (hereafter referred to as the 'speaker' and the 'robot') are in a room with two cones and a screen. The screen blocks the view of one of the cones from the speaker, but not the robot. Then, the speaker asks the robot to hand them the cone, using some locative clue such as "in front of me." If both of the cones match this description, then the robot should hand the speaker the cone that they know the speaker can see.

The model thus uses the ACT-R/S architecture in order to use spatial perspective taking to complete its task. In general, the ACT-R/S model fires a series of productions that inspect the viewpoint of the speaker (through the configural buffer) to determine which cone the speaker is referring to. Throughout the simulation, the model mentally walks and perceives from both the speaker's and the listener's viewpoint.[15]

3. Multimodal Interfaces in Space

We use a multimodal interface to process the various interactions with the robot. While there are a wide variety and many examples of multimodal interfaces, too numerous to cite here, there are a few multimodal interfaces that focus on the kinds of interactions with which we are concerned; namely, gestural and natural language modes of

interaction. For example, one gestural interface uses stylized gestures of arm and hand configurations[23] while another is limited to the use of gestural strokes on a PDA display.[24] The extent to which such devices will be employed in the environment of space has yet to be determined. However, since our interface has modules already in place, they are provided here for expository completeness. The use of a PDA device as currently envisioned might prove to be rather cumbersome for astronauts, already encumbered by a bulky spacesuit. However, future work may show that the use of such devices as wearable interfaces might prove beneficial for astronauts in certain situations.

Other interactive systems process information about the dialog using natural language input.[25,26] Our multimodal robot interface is unique in its combination of gestures and robust natural language understanding coupled with the capability of generating and understanding linguistic terms using spatial relations.

An interface supporting communication between a humanoid robot and a human sharing a real world environment and interacting with each other in that environment must include a natural language component. Just as humans interact with each other, not relying on monitors, joysticks or other computational devices with which to exchange information or request actions to be performed in the real world, humans interacting with humanoids will not necessarily use these devices in their daily interactions (although they may be provided for ancillary purposes). On the other hand, to facilitate communication, the interface will use natural language and gestures to allow the agents to communicate with each other in a very natural way. The emphasis here, of course, is on what is natural for the human, to provide a more habitable interface for the human to use. Thus, an interface which is to support collaboration between humans and humanoid robots must include a natural language component.

4. Understanding Language and Gestures

We currently employ a natural language interface that combines a ViaVoice™ speech recognition front-end with an in-house developed deep parsing system, NAUTILUS.[27] This gives the robot the capability to parse utterances, providing both syntactic representations and semantic interpretations. The semantic interpretation subsystem is integrated with other sensor and command inputs through use of a command interpretation system. The semantic interpretation, interpreted gestures from the vision system, and command inputs from the computer or other interfaces are compared, matched and resolved in the command interpretation system.

Using our multimodal interface (Figure 3), the human user can interact with a robot, using natural language and gestures. The natural language component of the interface embodied in the Spoken Commands and Command Interpreter modules of the interface uses ViaVoice™ to analyze spoken utterances. The speech signal is translated to a text string that is further analyzed by our natural language understanding system, NAUTILUS, to produce a regularized expression. This representation is linked, where necessary, to gesture information via the Gesture Interpreter, Goal Tracker/Spatial Relations component, and Appropriateness/Need Filter, and an appropriate robot action or response results.

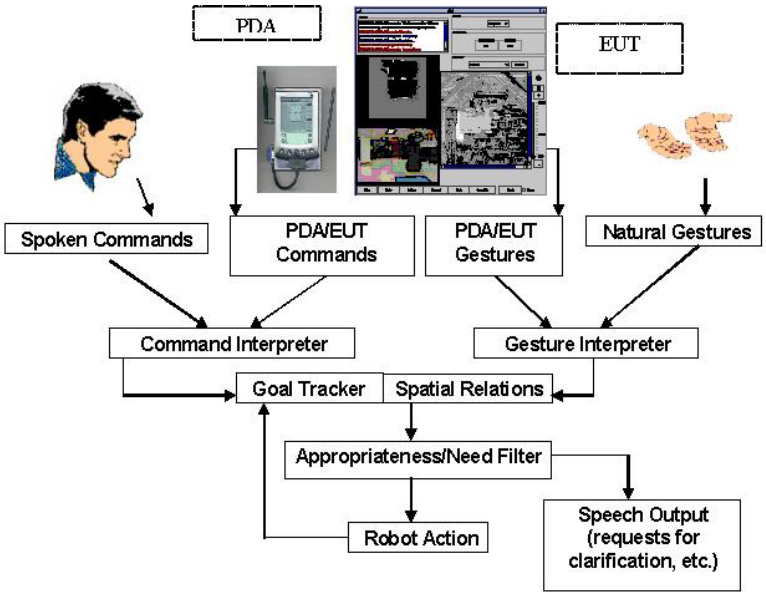

Fig. 3. Multimodal Interface for Human-Robot Collaboration.

For example, the human user can ask the robot "How many objects do you see?" ViaVoice™ analyzes the speech signal, producing a text string. NAUTILUS parses the string and produces a representation something like the following, simplified here for expository purposes.

```
(ASKWH                                                    (1)
  (MANY N3 (:CLASS OBJECT) PLURAL)
   (PRESENT #:V7791
        (:CLASS P-SEE)
        (:AGENT (PRON N1 (:CLASS SYSTEM) YOU))
        (:THEME N3)))
```

The parsed text string is mapped into a kind of semantic representation, shown here, in which the various verbs or predicates of the utterance (e.g. *see*) are mapped into corresponding semantic classes (*p-see*) that have particular argument structures (*agent*, *theme*). For example, "you" is the agent of the *p-see* class of verbs in this domain and "objects" is the theme of this verbal class, represented as "N3"—a kind of co-indexed trace element in the theme slot of the predicate, since this element is fronted in English wh-questions. If the spoken utterance requires a gesture for disambiguation (e.g. the sentence "Look over there"), the gesture components obtain and send the appropriate information to the Goal Tracker/Spatial Relations component where linguistic and gesture information are combined.

Both natural and so-called "symbolic" gestures are input to the multimodal interface. Users can gesture naturally by indicating directions, measurements, or specific locations with arm movements or they can use more symbolic gestures, by indicating paths and locations on a metric-map representation of the environment or video image on a PDA screen or end-user terminal (EUT). Users of this modality can point to locations and objects directly on the EUT monitor, thereby permitting the following kinds of utterances: "Go this way," "Pick up that object/wrench," or "Explore the area over there" using a real-time video display in addition to already available natural means of gestural interchange should the situation require its use. If the gesture — whatever its source — is valid, a message is sent to the appropriate robotics module(s) to generate the corresponding robot action. If the gesture is inappropriate, an error message is generated to inform the user, just as humans interact with other humans when further information is required or corrective action is needed for further exchanges. Where no gesture is required or is superfluous, the linguistic information maps directly to an appropriate robot command. In the above example (1), no further gesture information is required to understand the question about the number of objects seen.

In previous efforts we interacted with several non-humanoid mobile robots. As indicated earlier, as we focus more on working with humanoid robots, we believe natural gestures will become more prevalent in the kinds of interactions we study. Gesturing is a natural part of human-to-human communication. It disambiguates and provides information when no other means of communication is used. For example, we have already discussed the disambiguating nature of a gesture accompanying the utterance "Look over there." However, humans also gesture quite naturally and frequently as a non-verbal means of communicating information. Thus, a human worker collaborating with another worker in an assembly task might look in the direction of a needed tool and point at it. The co-worker will typically interpret this look and gesture as a combined non-verbal token indicating that the tool focused on and gestured at is needed, should be picked up and passed back to the first co-worker. In terms of the entire communicative act, both the look and the gesture indicate that a specific object is indicated, and the context of the interaction, namely assembly work, dictates that the object is somehow relevant to the current task and should therefore be obtained and handed over.

A verbal utterance might also accompany the foregoing non-verbal acts, such as "Get me that wrench" or simply "Hand me that." In the case of the first utterance, the object in the world has a location and a name. Its location is indicated by the deictic gestures perceived (head movement, eye gaze, finger pointing, etc.), but its name comes solely from the linguistic utterance. Whether or not the term "wrench" is already known by the second co-worker, the latter can locate the object and complete the task of handing it to the first co-worker. Further, even if the name of the object is not part of the second co-worker's lexicon, it can be inferred from the gestural context. Gestures have narrowed down the possibilities of what item in the world is known as a "wrench." In the case of the second utterance above, the name of the item is not uttered, but the item can still be retrieved and handed to the first co-worker. In this case, if the name of the item is unknown, the second co-worker can ask "What's this called?" as the co-worker passes the requested item.

We envision such interactions and behaviors as those outlined above as elements of possible scenarios between humans and Robonaut. Thus far, in our work on a multimodal interface to mobile robots, we have shown how various modes of our interface can be used to facilitate communication and collaboration. However, we

would like to extend such capabilities to a humanoid robot, as well as add learning, such as learning the name of an object previously unknown based on contextual (conversational and visual) information.

5. Spatial Reasoning

Building upon the existing framework of our natural language understanding system, and utilizing the on-board sensors for detecting objects, we are developing a spatial reasoning capability for the robot.[28,29,30,31] This spatial reasoning capability has been integrated with the natural language and gesture understanding modules through the use of a spatial modeling component based on the histogram of forces.[32] Force histograms are computed from a boundary representation of two objects to provide a qualitative model of the spatial relationship between the objects. Here, the histograms are computed between an environment object (extracted from sensory data) and the robot to produce an egocentric model of the robot's environment. Features extracted from the histograms are fed into a system of rules[33] or used as parameters in algorithms[30] to produce linguistic spatial terms. The spatial language component will be incorporated into the cognitive framework of the robot through a perspective-taking capability implemented using the Polyscheme architecture.

5.1. *Spatial language*

Spatial reasoning is important not only for solving complex navigation tasks, but also because we as human operators often think in terms of the relative spatial positions of objects, and we use relational linguistic terminology naturally in communicating with our human colleagues. For example, a speaker might say, "Hand me the wrench on the table." If the assistant cannot find the wrench, the speaker might say, "The wrench is to the left of the toolbox." The assistant need not be given precise coordinates for the wrench but can look in the area specified using spatial relational terms.

In a similar manner, this type of spatial language can be helpful for intuitive communication with a robot in many situations. Relative spatial terminology can be used to limit a search space by focusing attention in a specified region, as in "Look to the left of the toolbox and find the wrench." It can be used to issue robot commands, such as "Pick up the wrench on the table." A sequential combination of such directives can be used to describe and issue a high level task, such as, "Find the toolbox on the table behind you. The wrench is on the table to the left of the toolbox. Pick it up and bring it back to me." Finally, spatial language can also be used by the robot to describe its environment, thereby providing a natural linguistic description of the environment, such as "There is a wrench on the table to the left of the toolbox."

In all of these cases the use of spatial terminology increases the dynamic autonomy of the system by giving the human operator less restrictive and more natural language for communicating with the robot. However, the examples above also assume some level of object recognition by the robot.

To address the problem of linguistically identifying and understanding novel objects in the real world, the natural language understanding system interacts with the spatial relations component to assist in recognizing and labeling objects. This is achieved by allowing the user to dialog with the robot about novel objects. Once an object is labeled, the user can then issue additional commands using the spatial terms and referencing the named object. An example is shown below:

Human: "How many objects do you see?"
Robot: "I see 4 objects."
Human: "Where are they located?"
Robot: "There are two objects in front of me, one object on my right, and one object behind me."
Human: "The nearest object in front of you is a toolbox. Place the wrench to the left of the toolbox."

Establishing a common frame is necessary so that it is clear what is meant by spatial references generated both by the human operator as well as by the robot. Thus, if the human commands the robot, "Turn left," the robot must know whether the operator refers to the robot's left or the operator's left. In a human-robot dialog, if the robot places a second object "just to the left of the first object," is this the robot's left or the human's left? We anticipate that the dialoging, coupled with Polyscheme's ability to construct a world employing the objects under discussion, will produce an adequate representation of the perspectives of the various agents involved in the interaction.

Currently, commands using spatial references (e.g., "go to the right of the table") assume an extrinsic reference frame of the object (table) and are based on the robot's viewing perspective, consistent with Grabowski's "outside perspective".[34] That is, the spatial reference assumes the robot is facing the referent object. In the example, the robot would first turn to face the table, and then determine a target point to the right of the table.

There is some rationale for using the robot's viewing perspective. In human-robot experiments, Moratz et al. found that test subjects consistently used the robot's perspective when issuing commands.[35] We are investigating this by conducting human-factors experiments. In these experiments individuals who do not know the spatial reasoning capabilities and limitations of the robot provide instructions to the robot to perform various tasks requiring spatial referencing. The results of these studies[36,37] will be used to enhance the multimodal interface by establishing a common language for spatial referencing which incorporates those constructs and utterances most frequently used by untrained operators for commanding the robot.

Preliminary findings in our pilot study confirmed that human users usually take the robot's perspective when issuing directional or spatial commands when they have the robot's point of view. However, we also provided the human users with a God's-eye-view of the robot's environment. An interesting question arises when the user has a God's-eye-view of the robot's environment.[36] Namely, do humans continue to facilitate the interchange with addressee-centered spatial references or employ other types of perspectives, such as object-oriented or object-centric orientations to compensate for the

mismatch of perspectives? However, we cannot answer this question here because data analysis at this point is incomplete. Further findings are forthcoming.[36]

5.2. *Spatial representation*

In previous work we have used both 2D horizontal planes (e.g., an evidence grid map, built with range sensor data) and 2D vertical planes (using image data), but thus far they have not been combined. For Robonaut we will combine them to create a 2½D representation. To achieve the type of interaction described above, it is not necessary to build a full 3D, global representation of the environment. Rather, we assert that a more useful strategy is to obtain range information for a set of objects identified in the image plane. Human spatial language naturally separates the vertical and horizontal planes, e.g., "the wrench is on the table, vs. the wrench is to the left of the toolbox." Our linguistic representation provides a mapping for both locative prepositional phrases, e.g., "the wrench is on the table to the left of the toolbox." Processing the spatial information as two (roughly) orthogonal planes provides a better match with human spatial language.

Range information is extracted from stereo vision; the vision-based object recognition can assist in determining the correct correspondence between stereo images by constraining the region in the image. We do not need to label everything in the scene, but only those objects or landmarks that provide a basis to accomplish the robot's task.

5.3. *Spatial databases*

The position of recognized objects can be stored in a robot-centric frame such as the Sensory Ego Sphere (SES);[38] global position information is not necessary. The SES is a database implementation of Albus's proposed egosphere.[39] This spatial database provides an egocentric view of the world which is consistent with the robot's viewing perspective. The SES structure is a geodesic dome with a default frequency of 13, yielding a resolution of about five degrees with 1680 hexagonally-connected triangles. Each vertex can be labeled with an object identifier; some objects span multiple vertices. Objects may be retrieved using azimuth and elevation angles as indices into the database. An azimuth and elevation may also define a starting point in a search, for example, to look for an object in a specified region. In this case, a breadth-first search is executed using the specified azimuth and elevation as the starting node.[38]

As a database structure, sensory processes may be adding information to the database in the form of identified objects and ranges, while, at the same time, motor processes may be retrieving information for navigation or manipulation tasks. Being egocentric, SES facilitates sensor fusion and reasoning in the robot's inherent reference frame.[39]

In addition, the egocentric structure provides a convenient method of applying Previc's cognitive model which specifies how the egocentric space is subdivided and used depending on the type of task (manipulation, recognition, navigation, or

localization).[40] For example, peripersonal space is used for manipulation tasks, covering the central 60 degrees in the lower field of a body-centered egocentric space, up to about two meters. Navigation tasks use action space which covers a full 360 degrees with a range of about two meters up to intermediate distances.

We plan to evaluate the use of the SES as a spatial database for Robonaut. To achieve the type of spatial language dialog described above, we will project environment objects onto horizontal and vertical planes. For example, phrases such as "look for the wrench to the left of the toolbox" will utilize a horizontal plane at the appropriate height. The toolbox (stored in the SES with a known range) would be projected onto this horizontal plane and a region to the left of the toolbox, also located on the horizontal plane, would be computed. This left region location is then transformed back into the SES to define a starting node for the wrench search. In a similar way, a vertical plane can be used to form the relations above and below or on top.

6. Implementation

This section describes the integration of previously discussed concepts, speech understanding, and perspective-taking for the robotic astronaut assistant platform, Robonaut.

6.1. *Architecture*

The system implementing Robonaut's autonomy is made up of several modules (Figure 4). Modules communicate among each other using TCP/IP and/or Network Data Delivery Service (NDDS) version 2.3d.[41] The modules are implemented across multiple operating systems and machines. This section gives details of the implementation and functionality of each of the modules.

6.1.1. *Speech recognition and understanding*

Human speech is captured by microphone and then processed into a text string by ViaVoice™. The text string is then sent via TCP/IP socket connection to NAUTILUS. NAUTILUS parses the text and activates appropriate behaviors such as talking, movement, or reasoning via NLPProxy. ViaVoice™ is currently running under Windows XP while NAUTILUS has been compiled on Red Hat 7.2 kernel 2.4.2 under Allegro Lisp 6.1. Due to unavailability of a Linux version of NDDS and Windows license of Allegro Lisp, it was necessary to implement NLPProxy to pass messages from NAUTILUS to the remainder of the system. The proxy, a simple C server application, receives the command tokens from NAUTILUS over TCP/IP and sends out appropriate NDDS messages.

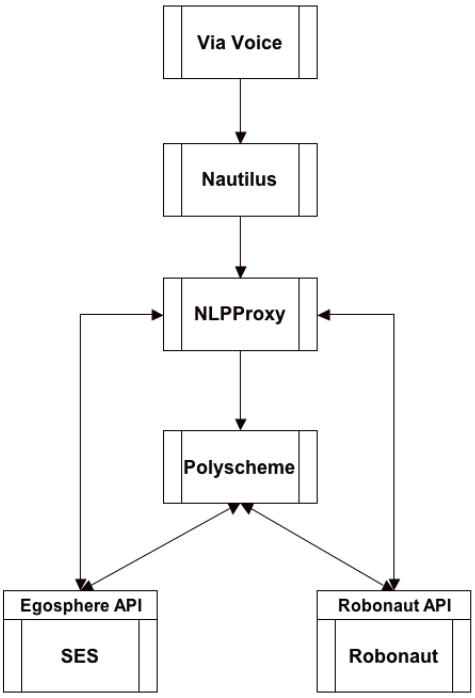

Fig. 4. Architecture diagram.

In addition to capabilities available from our previous work[28] such as simple motion commands (e.g., turn, move), spatial language terminology (between, to the left or right of, etc.) was adapted to match Robonaut's capabilities. The system was augmented to handle the concepts exemplified by verbs such as grasp, give, pick-up, push, press, point, and show to take advantage of Robonaut's arms and its grasping capabilities.

Less obviously, the collaborative nature of Robonaut's tasks also requires the system to handle words that structure the dialogues that such tasks typically require. This is being developed for parallel use in a robot that can learn the task from the human collaborator.[42] Task-structuring words include 'top-level' ways of announcing the task at hand (e.g., "Let's do task number 3"), sequencing words to arrange and delineate subtasks (e.g., now, first, next), and feedback (e.g., "That's wrong," "Okay," "Good").

6.1.2. Perspective-taking in Polyscheme

A perspective-taking specialist was implemented in Polyscheme. When Polyscheme receives a command from NAUTILUS (or rather NLPProxy) to reach for, give, or show an object, the perspective-taking specialist simulates a "world" from the human perspective based on knowledge about objects in the real world as recorded in Egosphere. It then reasons about relevant objects and resolves any possible ambiguities

in that world based on occlusions or proximity. Once the requested object has been found, the model issues motion commands to Robonaut. If the object is not found or the ambiguity cannot be resolved without additional information, the model provides the human with speech feedback.

Polyscheme and its specialists are implemented in Java and thus are platform independent. Since the specialist controls the robot directly, it makes use of Java Native Interface to call foreign functions implemented in Robonaut and Egosphere interfaces, which are currently only supported under Windows XP. Polyscheme currently runs under Windows XP with Java 1.4.2.

6.1.3. *Sensory Egosphere (SES)*

The SES is used to represent Robonaut's perception of its environment at its current state. Due to the difficulty of automatic object recognition, objects are currently identified and labeled by the human. Such objects can then be entered into the SES. The information obtained from the vision and the speech processes is stored in a local MYSQL database and includes the name of the object (nut1, driver2, etc.), its type (tool, person, part, etc.), frame of reference (chest or vision), its 3D position, and the pose of the robot at the time the object was identified. The Egosphere interface currently allows clients (e.g., Polyscheme, NLPProxy) to retrieve any attributes of a specific object and to return name of the object at a specified 3D location. The SES interface and database management application itself are implemented in C and C++ and run under Windows XP with MYSQL version 4.0.13.

6.1.4. *The robot: Robonaut*

As mentioned previously, Robonaut is a humanoid robot developed at NASA Johnson Space Center (JSC). There are currently two models. Robonaut Unit A has 43 degrees-of-freedom (DOF) as follows: a 2-DOF head, two 7-DOF arms, two 12-DOF hands, and a 3-DOF tail. The newer model, Robonaut Unit B, adds additional degrees of freedom in the head and tail and is currently mounted on a Segway platform providing it with additional mobility. Our work to date has been done on Unit A, but is completely compatible with Unit B.

Robonaut's interface allows clients (e.g., NLPProxy, Polyscheme) to easily control trajectories of motion of all of the robot's extremities as well as to obtain pose and sensor (e.g., vision, touch) information. Additional, higher-level functionality includes positioning of arms and grasping at desired locations, and looking toward a specified location. The Robonaut interface is written in C++ and currently runs under Windows XP. A simulator which supports most of Robonaut's functionality and includes a 3D display is provided as well.

6.2. *Demonstrations*

The following experimental scenarios were designed to present systems capabilities including speech recognition and natural language understanding, perspective-taking for resolution of occlusion- and proximity-based ambiguities among objects, object labeling, task learning, and autonomous grasping.

6.2.1. *Tool retrieval scenario*

In this scenario Robonaut assists a human by retrieving requested tools (Figure 5). There are a couple of objects of interest, in this case tools (or more specifically wrenches), which are placed in the environment, located between human and the robot. The human can see only one wrench because an obstacle, a plastic container, occludes the other one. Robonaut, on the other hand can see both wrenches. Robonaut is able to reason about the wrenches from human's perspective, giving it the ability to resolve the ambiguity in commands referring to the object, for example "Give me the wrench."

A perspective-taking specialist was implemented in Polyscheme which is able to resolve occlusion-based ambiguities and was integrated with speech recognition and natural language understanding modules, ViaVoice™ and NAUTILUS respectively. Furthermore, the model was able to get Robonaut to look toward and point at the appropriate object. Robonaut was also able to provide speech feedback during the interaction. In the current instance of the scenario, Robonaut was provided with a list of objects (e.g., wrenches, containers, people) and their positions in the environment. Integration efforts currently underway include visual object recognition provided by NASA JSC, SES by Vanderbilt University, and grasping implemented by the University of Massachusetts.

Fig. 5. Tool retrieval scenario.

6.2.2. *Wheel assembly task*

In this scenario Robonaut will fasten nuts onto a wheel (similar to lug nuts on an automobile wheel). There are several nuts available, a wheel, and several tools. A human identifies all the relevant objects in the environment through speech and gesture; for example, the human could say "Robonaut, this is a driver" while pointing to the driver in the environment. The human first shows Robonaut the task to give it the

opportunity to learn, and then the robot performs the task autonomously. During the learning phase of the task Robonaut assists the human by retrieving parts and appropriate tools.

This scenario will extend the integration performed for the tool retrieval scenario by taking advantage of the work being done at Vanderbilt University and Media Lab at MIT in learning by example and visual gesture tracking implemented at NASA JSC. The Polyscheme specialist will also be extended to allow Robonaut to resolve command ambiguities based upon objects' proximity to the person requesting them, in addition to resolving occlusion-based ambiguities as in the tool retrieval scenario.

7. Conclusions

Humanoid robots such as Robonaut offer many opportunities for advancing the use of robots in complex environments such as space, and for development of more effective interfaces for humans to interact with them. Once a sufficiently high level of interaction between robots and humans is achieved, the operation of and interaction with these robots will become less of an additional burden for the humans, and more of a collaboration to achieve the objectives of the task-at-hand. In this paper we describe our plans to endow Robonaut with cognitive capabilities which will support collaboration between human astronauts and Robonaut. We build upon our experience in natural language understanding, gesture recognition, spatial reasoning and cognitive modeling in achieving these goals.

Acknowledgements

Support for this effort was provided by the DARPA IPTO Mobile Autonomous Robot Software (DARPA MARS) Program. Thanks also to William Bluethmann, Mike Goza, and Rob Ambrose for their contributions to this effort.

References

1. M. A. Diftler, R. Platt, C. J. Culbert, R. O. Ambrose, W. J. Bluethmann, Evolution of the NASA/DARPA Robonaut Control System, in *Proceedings of the 2003 IEEE Conference on Robotics and Automation (ICRA),* (IEEE, Taipei, Taiwan, 2003).

2. G. A. Miller and P. H. Johnson-Laird. *Language and Perception*, (Harvard University Press, 1976).

3. B. Tversky, "Cognitive maps, cognitive collages, and spatial mental model," in A. U. Frank and I. Campari (Eds.), *Spatial information theory: Theoretical basis for GIS* (Springer-Verlag, 1993).

4. M. Bugajska, A. Schultz, T. J. Trafton, M. Taylor, and F. Mintz, "A Hybrid Cognitive-Reactive Multi-Agent Controller," in *Proceedings of 2002 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS-2002),* (IEEE, EPFL, Switzerland, 2002).

5. J. G. Trafton., A. Schultz, D. Perzanowski, W. Adams, M. Bugajska, N. L. Cassimatis, and D. Brock, "Children and robots learning to play hide and seek," in *Proceedings of the IJCAI Workshop on Cognitive Modeling of Agents and Multi-Agent Interaction, (*Acapulco, Mexico, 2003).

6. J. R. Anderson and C. Lebiere. *The atomic components of thought* (Lawrence Erlbaum, 1998).

7. N. L. Cassimatis., Polyscheme: A cognitive architecture for integrating multiple representation and inference schemes, PhD dissertation, (MIT Media Laboratory, 2002).

8. E. M. Altmann and J. G. Trafton, An activation-based model of memory for goals. In *Cognitive Science*, pp. 39-83 (2002).

9. J. R. Anderson, M. Matessa, and C. Lebiere. ACT-R: A theory of higher level cognition and its relation to visual attention, in *Human-Computer Interaction*, 12 (4), pp. 439-462 (ASME Press, 763-768, 1997).

10. C. D. Schunn and J. R. Anderson, Scientific Discovery, in J. R. Anderson and C. Lebiere (Eds.), *Atomic Components of Thought* (Lawrence Erlbaum, 1998).

11. J. Huttenlocher and L Kubicek, The coding and transformation of spatial information. *Cognitive Psychology, 11*, pp. 375-394 (1979).

12. N. Newcombe and J. Huttenlocher, Children's early ability to solve perspective taking problems. *Developmental Psychology, 28*, pp. 654-664 (1992).

13. R. Wallace, K. L. Allan, and C. T. Tribol, Spatial perspective-taking errors in children. *Perceptual and Motor Skills, 92*(3), pp. 633-639 (2001).

14. N. L. Cassimatis, J. G. Trafton, A. Schultz, and M. Bugajska. Integrating Cognition, Perception and Action through Mental Simulation in Robots. In *Proceedings of the 2004 AAAI Spring Symposium on Knowledge Representation and Ontology for Autonomous Systems*: (AAAI, 2004).

15. L. M. Hiatt, J. G. Trafton, A. Harrison, and A. Schultz, Using similar representations to improve human-robot interaction, in Trafton, J. G., Schultz, A. C., Cassimatis, N. L., Hiatt, L. M., Perzanowski, D., Brock, D. P., et al. (in press).

16. J. G. Trafton, S. B. Trickett, and F. E. Mintz, Connecting Internal and External Representations: Spatial Transformations of Scientific Visualizations. *Foundations of Science* (in press).

17. http://simon.lrdc.pitt.edu/~harrison/jactr.html

18. http://simon.lrdc.pitt.edu/~harrison/

19. R. F. Wang and E. S. Spelke, Human spatial representation: Insights from animals. *Trends in Cognitive Sciences, 6*(9), pp. 376-382 (2002).

20. J. G. Trafton, S. Marshall, F. E. Mintz, and S. B. Trickett, Extracting explicit and implicit information from complex visualizations, in M. Hegarty, B. Meyer & H. Narayanan (Eds.), *Diagramatic representation and inference,* pp. 206-220 (Berlin Heidelberg: Springer-Verlag, 2002).

21. M. D. Byrne and J. R. Anderson, Perception and action. in J. R. Anderson & C. Lebiere (Eds.), *Atomic Components of thought,* pp. 167-200, (Mahwah, NJ: Lawrence Erlbaum, 1998).

22. A. M. Harrison and C. D. Schunn, ACT-R/S: Look Ma, No "Cognitive-map"! in *International Conference on Cognitive Modeling* (2003).

23. D. Kortenkamp, E. Huber, and P. Bonasso, Recognizing and Interpreting Gestures on a Mobile Robot," in *Proceedings of AAAI* (1996).

24. T. W. Fong, F. Conti, S. Grange and C. Baur, Novel Interfaces for Remote Driving: Gesture, haptic, and PDA, in SPIE 4195-33*, SPIE Telemanipulator and Telepresence Technologies VII* (2000).

25. C. Rich, C. L. Sidner, and N. Lesh, COLLAGEN: Applying collaborative discourse theory to human-computer interaction, in *AI Magazine,* vol. 22, no. 4, pp. 15-25 (2001).

26. J. F. Allen, D. K. Byron, M, Dzikovska, G. Ferguson, L. Galescu and A. Stent, Toward conversational human-computer interaction, in *AI Magazine,* vol. 22, no. 4, pp. 27-37 (2001).

27. K. Wauchope, Eucalyptus: Integrating Natural Language Input with a Graphical User Interface, Naval Research Laboratory Technical Report NRL/FR/5510-94-9711 (Washington, DC, 1994).

28. M. Skubic, D. Perzanowski, A. Schultz, and W. Adams, Using Spatial Language in a Human-Robot Dialog, in *Proceedings of the IEEE 2002 International Conference on Robotics and Automation*, pp. 4143-4148 (IEEE, 2002).

29. M. Skubic, D. Perzanowski, S. Blisard, A. Schultz, W. Adams, M. Bugajska and D. Brock, Spatial Language for Human-Robot Dialogs, in *IEEE Transactions on SMC*, Part C, Special Issue on Human-Robot Interaction (IEEE, 2003).

30. M. Skubic and S. Blisard, Go to the Right of the Pillar: Modeling Unoccupied Regions for Robot Directives, in *2002 AAAI Fall Symposium, Human-Robot Interaction Workshop*, AAAI Technical Report FS-02-03 (2002).

31. M. Skubic, P. Matsakis, G. Chronis and J. Keller, Generating Multi-Level Linguistic Spatial Descriptions from Range Sensor Readings Using the Histogram of Forces, in *Autonomous Robots*, vol. 14, no. 1, pp. 51-69 (2003).

32. P. Matsakis and L. Wendling, A New Way to Represent the Relative Position of Areal Objects, in *IEEE Pattern Analysis and Machine Intelligence*, vol. 21, no. 7, pp. 634-643 (1999).

33. P. Matsakis, J. Keller, L. Wendling, J. Marjamaa, and O. Sjahputera, Linguistic Description of Relative Positions of Objects in Images, in *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 31, No. 4, pp. 573-588 (IEEE, 2001).

34. J. Grabowski, A Uniform Anthropomorphological Approach to the Human Conception of Dimensional Relations, in *Spatial Cognition and Computation*, vol. 1, pp. 349-363 (1999).

35. R. Moratz, K. Fischer and T. Tenbrink, Cognitive Modeling of Spatial Reference for Human-Robot Interaction, in *International Journal on Artificial Intelligence Tools*, vol. 10, no. 4, pp. 589-611 (2001).

36. D. Perzanowski, W. Adams, M. Bugajska, S. Thomas, D. Brock, D. Sofge, S. Blisard, A. Schultz, and J. G. Trafton, Toward Multimodal Cooperation and Collaboration between Humans and Robots, in *Proceedings of the AIAA 1st Intelligent Systems Technical Conference*, Chicago, IL (2004).

37. D. Perzanowski, D. Brock, S. Blisard, W.Adams, M.Bugajska, A. Schultz, G. Trafton, M. Skubic, Finding the FOO: A Pilot Study for a Multimodal Interface, in *Proceedings of the IEEE Systems, Man, and Cybernetics Conference*, pp. 3218-3223 (Washington, DC, 2003).

38. R. A. Peters II, K. Hambuchen, K. Kawamura, and D. M. Wilkes, The Sensory Ego-Sphere as a Short-Term Memory for Humanoids, in *Proceedings of the IEEE-RAS International Conference on Humanoid Robots* (IEEE, 2001).

39. J. S. Albus, Outline for a theory of intelligence, in *IEEE Trans. Systems, Man and Cybernetics*, vol. 21, no. 3 (IEEE, 1991).

40. F. H. Previc, The Neuropsychology of 3-D Space, *Psychological Review*, vol. 124, no. 2, pp. 123-164 (1998).

41. RTI NDDS. http://www.rti.com/products/ndds/index.html (April 2004).

42. C. Breazeal, G. Hoffman and A. Lockerd, Teaching and Working with Robots as a Collaboration, in *Proceedings of the Third International Joint Conference on Autonomous Agents and Multi-Agent Systems – Volume 3*, pp. 1030-1037 (2004).